

# Package ‘latex2exp’

May 8, 2026

**Type** Package

**Title** Use LaTeX Expressions in Plots

**Version** 0.9.8

**Date** 2024-07-05

**Description** Parses and converts LaTeX math formulas to R's plotmath expressions, used to enter mathematical formulas and symbols to be rendered as text, axis labels, etc. throughout R's plotting system.

**License** MIT + file LICENSE

**URL** <https://www.stefanom.io/latex2exp/>,  
<https://github.com/stefano-meschiari/latex2exp>

**BugReports** <https://github.com/stefano-meschiari/latex2exp/issues>

**Encoding** UTF-8

**Suggests** tools, testthat, waldo, knitr, ggplot2, rmarkdown, purrr,  
tibble, reactable, htmltools, RCurl, rlang, dplyr

**VignetteBuilder** knitr

**RoxygenNote** 7.2.3

**Language** en-US

**NeedsCompilation** no

**Author** Stefano Meschiari [aut, cre]

**Maintainer** Stefano Meschiari <[stefano.meschiari@gmail.com](mailto:stefano.meschiari@gmail.com)>

**Repository** CRAN

**Date/Publication** 2026-01-09 09:00:29 UTC

## Contents

latex2exp . . . . .	2
latex2exp_examples . . . . .	2
latex2exp_supported . . . . .	3
plot.expression . . . . .	3

print.latexpexpression . . . . .	4
print.latextoken2 . . . . .	4
render_latex . . . . .	5
TeX . . . . .	5

<b>Index</b>	<b>7</b>
--------------	----------

---

latex2exp	<i>Deprecated; use <a href="#">TeX</a> instead.</i>
-----------	---

---

### Description

Deprecated; use [TeX](#) instead.

### Usage

```
latex2exp(string, output = c("expression", "character", "ast"))
```

### Arguments

string	A character vector containing LaTeX expressions. Note that any backslashes must be escaped (e.g. "\$\alpha").
output	The returned object, one of "expression" (default, returns a plotmath expression ready for plotting), "character" (returns the expression as a string), and "ast" (returns the tree used to generate the expression).

### Value

Returns an expression (see the output parameter).

---

latex2exp_examples	<i>latex2exp Examples</i>
--------------------	---------------------------

---

### Description

Plots a number of example LaTeX string, as parsed by [TeX](#).

### Usage

```
latex2exp_examples(cex = 1)
```

### Arguments

cex	Multiplier for font size
-----	--------------------------

---

latex2exp\_supported     *Returns the list of supported LaTeX commands.*

---

### Description

If show is TRUE, also show a searchable table of symbols.

### Usage

```
latex2exp_supported(show = FALSE, ...)
```

### Arguments

show	Show a searchable table of symbols
...	Other parameters (not used)

### Value

A data frame containing a table of supported LaTeX commands.

---

plot.expression     *Previews a LaTeX equation*

---

### Description

Plots the result of a call to [TeX] on the current graphical device. This is useful to preview the output before placing it on a plot.

### Usage

```
## S3 method for class 'expression'
plot(x, ..., main = NULL)
```

### Arguments

x	A <a href="#">plotmath</a> expression returned by <a href="#">TeX</a> .
...	Parameters to be passed to the <a href="#">text</a> function.
main	Title of the plot

### Examples

```
plot(TeX("Example equation: $a \\geq b$"))
```

---

```
print.latexexpression Print an expression returned by TeX()
```

---

### Description

Prints out the plotmath expression generated by [TeX](#) and the original TeX string.

### Usage

```
## S3 method for class 'latexexpression'
print(x, ...)
```

### Arguments

x	Object to print
...	Ignored

---

```
print.latextoken2 Prints out a parsed LaTeX object, as returned by TeX(..., output = 'ast'). This is primarily used for debugging.
```

---

### Description

Prints out a parsed LaTeX object, as returned by `TeX(..., output = 'ast')`. This is primarily used for debugging.

### Usage

```
## S3 method for class 'latextoken2'
print(x, depth = 0, ...)
```

### Arguments

x	The object
depth	Increases padding when recursing down the parsed structure
...	(Ignored)

---

render_latex	<i>Renders a LaTeX tree</i>
--------------	-----------------------------

---

**Description**

Returns a string that is a valid plotmath expression, given a LaTeX tree returned by parse\_latex.

**Usage**

```
render_latex(tokens, user_defined = list(), hack_parentheses = FALSE)
```

**Arguments**

tokens	tree of tokens
user_defined	any custom definitions of commands passed to <a href="#">TeX</a>
hack_parentheses	render parentheses using <code>group('(', phantom(), ' .')</code> and <code>group(')', phantom(), ' .')</code> . This is useful to return valid expressions when the LaTeX source contains mismatched parentheses, but makes the returned expression much less tidy.

**Value**

String that should be parseable as a valid plotmath expression

---

TeX	<i>Converts LaTeX to a <a href="#">plotmath</a> expression.</i>
-----	---

---

**Description**

TeX converts a string comprising LaTeX commands (such as a math equation) to a [plotmath](#) expression. Plotmath expressions can be used through R's graphic system to represent formatted text and equations.

**Usage**

```
TeX(
  input,
  bold = FALSE,
  italic = FALSE,
  user_defined = list(),
  output = c("expression", "character", "ast")
)
```

**Arguments**

<code>input</code>	A character vector containing LaTeX strings. Note that any backslashes must be escaped (e.g. <code>"\$alpha"</code> ).
<code>bold</code>	Whether to make the entire label bold
<code>italic</code>	Whether to make the entire label italic
<code>user_defined</code>	Described in the "Adding New Commands" section.
<code>output</code>	The returned object, one of "expression" (default, returns a plotmath expression ready for plotting), "character" (returns the expression as a string), and "ast" (returns the tree used to generate the expression).

**Value**

Returns a plotmath expression by default. The `output` parameter can modify the type of the returned value.

If more than one string is specified in the `input` parameter, returns a list of expressions.

**Adding new commands**

New LaTeX commands can be defined by supplying the `user_defined` parameter. The `user_defined` parameter is a list that contains LaTeX commands as names, and template strings as values. A LaTeX command that matches one of the names is translated into the corresponding string and included in the final plotmath expression. The file `symbols.R` in the source code of this package contains one such table that can be used as a reference.

The template string can include one of the following special template parameters:

- `$arg1`, `$arg2`, ... represent the first, second, ... brace argument. E.g. for `\frac{x}{y}`, `$arg1` is `x` and `$arg2` is `y`.
- `$opt` is an optional argument in square brackets. E.g. for `\sqrt[2]{x}`, `$opt` is `2`.
- `$sub` and `$sup` are arguments in the exponent (^) or subscript (\_) following the current expression. E.g. for `\sum^x`, `$sup` is `x`.
- `$LEFT` and `$RIGHT` are substituted the previous and following LaTeX expression relative to the current token.

See the Examples section for an example of using the `user_defined` option.

**Examples**

```
TeX("$\\alpha$") # plots the greek alpha character
TeX("The ratio of 1 and 2 is $\\frac{1}{2}$")

a <- 1:100
plot(a, a^2, xlab = TeX("$\\alpha$"), ylab = TeX("$\\alpha^2$"))

# create a \\variance command that takes a single argument
TeX("$\\variance{X} = 10$",
  user_defined = list("\\variance" = "sigma[$arg1]^2"))
```

# Index

latex2exp, [2](#)  
latex2exp\_examples, [2](#)  
latex2exp\_supported, [3](#)

plot.expression, [3](#)  
plotmath, [3](#), [5](#)  
print.latexpression, [4](#)  
print.latextoken2, [4](#)

render\_latex, [5](#)

TeX, [2-5](#), [5](#)  
text, [3](#)